AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1-15. (cancelled)

16. (currently amended): A network interface card comprising:

an upper layer protocol (ULP) handler;

a TCP handler capable of interfacing with said ULP handler; and,

a link handler,

wherein the ULP handler of the network interface card is adapted to take over and perform at least one session layer function of a host computer connected to a network, said network interface card is capable of receiving commands from an enhanced stack belonging to said host, said enhanced stack being further capable of supporting session layer acceleration, the commands between said enhanced stack and said network interface card are performed using acceleration primitives, and ~~The network interface card of claim 15, wherein~~ said network interface card is operable to handles only a subset said acceleration primitives sent to said network interface card from a plurality of said acceleration primitives sent to a plurality of network interface card devices.

17-19. (cancelled):

20. (currently amended): A network interface card comprising:

an upper layer protocol (ULP) handler;

a TCP handler capable of interfacing with said ULP handler; and,

a link handler,

wherein the ULP handler of the network interface card is adapted to take over and perform at least one session layer function of a host computer connected to a network, said network interface card is capable of receiving commands from an enhanced stack belonging to said host, said enhanced stack being further capable of supporting session layer acceleration, the commands between said enhanced stack and said network interface card are performed using acceleration primitives, at least one of said acceleration primitives is used to establish a direct connection between ULP of said host and said ULP handler, said direct connection enables at least one function associated with a TCP/IP layer to be processed on said network interface card, said direct connection enables transferring data to said network interface card from said host and transferring data from said network interface card to said host, and ~~The network interface card of claim 19, wherein~~ said transferring data to said network interface card includes at least one of a transfer in request, a success transfer in reply and a fail transfer in reply.

21. (currently amended): A network interface card comprising:

an upper layer protocol (ULP) handler;

a TCP handler capable of interfacing with said ULP handler; and,

a link handler,

wherein the ULP handler of the network interface card is adapted to take over and perform at least one session layer function of a host computer connected to a network, said network interface card is capable of receiving commands from an enhanced stack belonging to said host, said enhanced stack being further capable of supporting session layer acceleration, the commands between said enhanced stack and said network interface card are performed using acceleration primitives, at least one of said acceleration primitives is used to establish a direct connection between ULP of said host and said ULP handler, said direct connection enables at least one function associated with a TCP/IP layer to be processed on said network interface card, said direct connection enables transferring data to said network interface card from said host and transferring data from said network interface card to said host, and ~~The network interface card of claim 19, wherein~~ said transferring data from said network interface card includes at least one of a transfer out request, a success transfer out reply and a fail transfer out reply.

22. (cancelled):

23. (currently amended): A network interface card comprising:

an upper layer protocol (ULP) handler;

a TCP handler capable of interfacing with said ULP handler; and,

a link handler,

wherein the ULP handler of the network interface card is adapted to take over and perform at least one session layer function of a host computer connected to a network, said network interface card is capable of receiving commands from an enhanced stack belonging to

said host, said enhanced stack being further capable of supporting session layer acceleration, the commands between said enhanced stack and said network interface card are performed using acceleration primitives, said acceleration primitives are enabled by the use of an application programming interface (API) for interfacing between said host and said network interface card, said API being further comprised of a plurality of acceleration primitives, and ~~The network interface card of claim 22, wherein~~ at least one of said ~~message~~ acceleration primitives is a connection transfer in ~~message~~ acceleration primitive sent in order to transfer a connection for acceleration by the network interface card.

24. (previously presented): The network interface card of claim 23, wherein said connection transfer in acceleration primitive further contains at least information required to start a new or continued processing of an existing connection.

25. (original): The network interface card of claim 24, wherein said necessary information comprises at least one of a connection 4tuple, initial sequence number, unacknowledged sequence number, acknowledgement sequence number, current sent time stamp, current received timestamp and remote or local negotiated window scale values.

26. (previously presented): The network interface card of claim 23, wherein said connection transfer in acceleration primitive includes a unique identifier for connection reference, said unique identifier being further recognizable by said network interface card and by said host.

27. (previously presented): The network interface card of claim 22 wherein at least one of said acceleration primitives is a connection transfer out acceleration primitive capable of causing said network interface card to transfer a connection out of said network interface card.

28. (previously presented): The network interface card of claim 27, wherein said connection transfer out acceleration primitive provides only a connection reference to said network interface card.

29. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection disconnect acceleration primitive, said acceleration primitive being capable of causing said network interface card to gradually close a previously opened connection to said network interface card.

30. (previously presented): The network interface card of claim 29, wherein said connection disconnect acceleration primitive provides only a connection reference to said network interface card.

31. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection abort acceleration primitive, said acceleration primitive being capable of causing said network interface card to abort a previously opened connection to said network interface card.

32. (previously amended): The network interface card of claim 31, wherein said connection abort acceleration primitive provides only a connection reference to said network interface card.

33. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection shutdown of transmission acceleration primitive, said acceleration primitive being capable of causing said network interface card to gracefully close a write side of a connection of said NIC.

34. (previously presented): The network interface card of claim 33, wherein said connection shutdown of transmission acceleration primitive provides only a connection reference to said network interface card.

35. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection send acceleration primitive, said acceleration primitive being capable of causing a transmission of data over an established connection.

36. (previously presented): The network interface card of claim 35, wherein said connection send acceleration primitive is associated with data related to a TCP/IP connection, said data is at least one of connection reference, list of buffers in the host memory and their length, said buffers containing data to be transferred over a connection.

37. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection receive acceleration primitive, said acceleration primitive being capable of causing a reception of data over an established connection, the date being received by connection specific receive buffers in a host memory.

38. (previously presented): The network interface card of claim 37, wherein said connection receive acceleration primitive is associated with data related to a TCP/IP connection, said date is at least one of connection reference, list of connection specific buffers in the host memory and their length.

39. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection synchronization acceleration primitive, said acceleration primitive being capable of flushing existing message pipes between the host and said network interface card.

40. (previously presented): The network interface card of claim 39, wherein said connection synchronization acceleration primitive further includes a connection reference.

41. (previously presented): The network interface card of claim 39, wherein a connection synchronization reply acceleration primitive is sent in response to said connection synchronization acceleration primitive.

42. (previously presented): The network interface card of claim 41, wherein said connection synchronization reply acceleration primitive further includes a connection reference.

43. (previously presented): The network interface card of claim 22, wherein said network interface card is capable of sending a connection send notify acceleration primitive, wherein said connection send notify acceleration primitive notifies of a successful transfer of an amount of data related to an offloaded TCP connection.

44. (previously presented): The network interface card of claim 43, wherein data associated with said connection send notify acceleration primitive includes at least one of a connection reference and amount of data successfully transferred over the connection.

45. (previously presented): The network interface card of claim 22, wherein at least one of said acceleration primitives is a connection receive notify acceleration primitive, said acceleration primitive being capable of notifying of the reception of additional data by said network interface card over a connection.

46. (original): The network interface card of claim 45, wherein said additional data may be directed to one of an anonymous host buffer and a connection specific host buffer.

47. (previously presented): The network interface card of claim 46, wherein at least one of said acceleration primitives is a asynchronous buffer acceleration primitive, said acceleration

primitive being capable of posting said anonymous receive buffers to said network interface card.

48. (original): The network interface card of claim 47, wherein said anonymous receive buffers are used for a received TCP data and a layer 2 data.

49. (previously presented): The network interface card of claim 47, wherein data associated with said asynchronous buffer acceleration primitive includes a list of buffers in host memory and buffer lengths.

50. (previously presented): The network interface card of claim 45, wherein data associated with said connection receive notify acceleration primitive includes connection reference, buffer identification and amount of data posted into the buffer.

51. (original): The network interface card of claim 22, wherein said network interface card is capable of providing a notification from said network interface card to the host with an indication of a change in connection state.

52. (original): The network interface card of claim 51, wherein the data associated with said notification includes connection reference, notification type and a connection state.

53. (original): The network interface card of claim 52, wherein said notification type include connection established, connection disconnected, connection timed-out and connection gracefully closed.

54.-60. (cancelled)

61. (original): A method for acceleration of a session layer network operation, said method comprising:

a) sending a sequence of initialization commands from a ULP driver of a host to transport accelerator provider (TAP) of said host;

b) sending a transfer message from said TAP to a TCP handler of a network interface card (NIC);

c) sending from said NIC a synchronization command to a server over a network connecting said host computer and said server;

d) receiving by said NIC a synchronization acknowledgement message over said network from said server;

e) sending from said NIC an acknowledgment message to said server;

f) sending a notification command to a ULP handler of said NIC;

g) sending from said NIC a connection notification command to said TAP of said host; and,

h) sending a connected information command to said ULP driver of said host.

62. (previously presented): The method of claim 61, wherein said method further comprises using an application programming interface (API) for interfacing between said host and said L5NIC, said API being further comprised of a plurality of acceleration primitives.

63. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection transfer in acceleration primitive sent in order to transfer a connection for acceleration by the network interface card.

64. (previously presented): The method of claim 63, wherein said connection transfer in acceleration primitive further contains at least information required to start a new or continued processing of an existing connection.

65. (original): The method of claim 64, wherein said necessary information comprises at least one of a connection 4tuple, initial sequence number, unacknowledged sequence number, acknowledgement sequence number, current sent time stamp, current received timestamp and remote or local negotiated window scale values.

66. (previously presented): The method of claim 63, wherein said connection transfer in acceleration primitive includes a unique identifier for connection reference, said unique identifier being further recognizable by said network interface card and by said host.

67. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection transfer out acceleration primitive capable of causing said network interface card to transfer a connection out of said network interface card.

68. (previously presented): The method of claim 67, wherein said connection transfer out acceleration primitive provides only a connection reference to said network interface card.

69. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection disconnect acceleration primitive, said acceleration primitive being capable of causing said network interface card to gradually close a previously opened connection to said network interface card.

70. (previously presented): The method of claim 69, wherein said connection disconnect acceleration primitive provides only a connection reference to said network interface card.

71. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection abort acceleration primitive, said acceleration primitive being capable of causing said network interface card to abort a previously opened connection to said network interface card.

72. (previously presented): The method of claim 71, wherein said connection abort acceleration primitive provides only a connection reference to said network interface card.

73. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection shutdown of transmission acceleration primitive, said acceleration primitive being capable of causing said network interface card to gracefully close a write side of a connection of said NIC.

74. (previously presented): The method of claim 73, wherein said connection shutdown of transmission acceleration primitive provides only a connection reference to said network interface card.

75. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection send acceleration primitive, said acceleration primitive being capable of causing a transmission of data over an established connection.

76. (previously presented): The method of claim 75, wherein said connection send acceleration primitive is associated with data related to a TCP/IP connection, said data is at least one of connection reference, list of buffers in the host memory and their length, said buffers containing data to be transferred over a connection.

77. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection receive acceleration primitive, said acceleration primitive

being capable of causing a reception of data over an established connection, the date being received by connection specific receive buffers in a host memory.

78. (previously presented): The method of claim 77, wherein said connection receive acceleration primitive is associated with data related to a TCP/IP connection, said date is at least one of connection reference, list of connection specific buffers in the host memory and their length.

79. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection synchronization acceleration primitive, said acceleration primitive being capable of flushing existing message pipes between the host and said network interface card.

80. (previously presented): The method of claim 79, wherein said connection synchronization acceleration primitive further includes a connection reference.

81. (previously presented): The method of claim 79, wherein a connection synchronization reply acceleration primitive is sent in response to said connection synchronization acceleration primitive.

82. (previously presented): The method of claim 81, wherein said connection synchronization reply acceleration primitive further includes a connection reference.

83. (previously presented): The method of claim 62, wherein said network interface card is capable of sending a connection send notify acceleration primitive, wherein said connection send notify acceleration primitive notifies of a successful transfer of an amount of data related to an offloaded TCP connection.

84. (previously presented): The method of claim 83, wherein data associated with said connection send notify acceleration primitive includes at least one of a connection reference and amount of data successfully transferred over the connection.

85. (previously presented): The method of claim 62, wherein at least one of said acceleration primitives is a connection receive notify acceleration primitive, said acceleration primitive being capable of notifying of the reception of additional data by said network interface card over a connection.

86. (original): The method of claim 85, wherein said additional data may be directed to one of an anonymous host buffer and a connection specific host buffer.

87. (previously presented): The method of claim 86, wherein at least one of said acceleration primitives is a asynchronous buffer acceleration primitive, said acceleration primitive being capable of posting said anonymous receive buffers to said network interface card.

88. (original): The method of claim 87, wherein said anonymous receive buffers are used for a received TCP data and a layer 2 data.

89. (previously presented): The method of claim 87, wherein data associated with said asynchronous buffer acceleration primitive includes a list of buffers in host memory and buffer lengths.

90. (previously presented): The method of claim 85, wherein data associated with said connection receive notify acceleration primitive includes connection reference, buffer identification and amount of data posted into the buffer.

91. (original): The method of claim 62, wherein said network interface card is capable of providing a notification from said network interface card to the host with an indication of a change in connection state.

92. (original): The method of claim 91, wherein the data associated with said notification includes connection reference, notification type and a connection state.

93. (original): The method of claim 92, wherein said notification type include connection established, connection disconnected, connection timed-out and connection gracefully closed.

94.-134. (cancelled)

135. (original): The method of claim 61, wherein the network interface card corresponds to a layer 5 of a seven layer OSI network architecture.